Extending the C2 Lab Environment: Integrating Proxies, DNS Infrastructure, and a Registered Domain

# 1. Introduction: Enhancing C2 Lab Realism and Capability

A well-designed Command and Control (C2) lab environment is paramount for cybersecurity professionals seeking to understand adversary tactics, develop robust defenses, and evaluate security tools. Such environments allow for the safe experimentation with offensive techniques, providing invaluable insights into the complexities of modern cyber threats. However, basic C2 setups often lack the sophistication required to accurately simulate real-world scenarios. Threat actors frequently employ advanced infrastructure components like proxy servers, custom Domain Name System (DNS), and registered domains to enhance their operations, providing anonymity, evading detection, and maintaining persistent communication channels.

To overcome the limitations of rudimentary C2 labs, this report outlines a comprehensive plan for extending an existing environment with these critical components. The integration of proxy servers will enable the simulation of attacker anonymity and traffic redirection. Implementing a custom DNS infrastructure will allow for the exploration of covert communication techniques such as DNS tunneling and beaconing, while a registered domain will add a layer of realism to the simulated C2 operations and facilitate the study of domain reputation-based detection. This report will delve into the research behind each component, provide justifications for the chosen solutions, detail the setup and configuration processes, and discuss the security implications from both offensive and defensive perspectives, including relevant mitigation strategies. The ultimate objective is to provide a detailed and actionable plan that empowers cybersecurity professionals to create a more realistic and capable C2 lab environment for training, research, and development purposes.

# 2. Proxy Server Integration

## Purpose and Types of Proxy Servers in C2 Infrastructure

Proxy servers act as intermediaries in network communication, forwarding requests from clients to servers and vice versa. In the context of C2 infrastructure, they play a crucial role in obfuscating the attacker's origin and evading defensive measures. Several types of proxy servers are relevant to C2 operations. **Forward proxies** are utilized by the attacker's infrastructure to route traffic to target systems. By doing so, they can mask the originating IP address of the C2 server, making it more difficult to trace the attack back to its source 1. This added layer of indirection provides a degree of anonymity for the attacker's C2 server, complicating attribution efforts.

**Reverse proxies**, on the other hand, are deployed in front of the C2 server itself 2. Their purpose is to hide the true location of the C2 server, potentially offering benefits such as load balancing and Secure Sockets Layer (SSL) termination. This abstraction enhances the resilience of the C2 infrastructure, as the backend server can be changed without altering the publicly facing IP address in case of compromise.

**Anonymizing proxies** are specifically designed to obscure the identity of the user or the server by removing or altering identifying information within network traffic 1. Different levels of anonymization exist; basic anonymous proxies might only conceal the IP address, while **high anonymity proxies** take further steps to remove or modify headers and other data that could reveal the originator's identity.

To further complicate tracking, **rotating proxies** assign different IP addresses to users or servers over time 1. This dynamic IP allocation makes it significantly harder to track activity based on IP addresses and increases the difficulty of IP-based blocking of C2 communications. If one IP address is blacklisted, the system can quickly switch to a new, unblocked one.

For a strong emphasis on anonymity, attackers may leverage **TOR proxies**. TOR (The Onion Router) routes traffic through a globally distributed network of volunteer-operated servers, encrypting data in multiple layers at each hop 1. This multi-layered encryption and routing mechanism provides a high level of anonymity, although it can sometimes impact the speed of the connection. **I2P proxies** offer a similar anonymous network, focusing on anonymity within the I2P (Invisible Internet Project) network itself 1. While less widely used than TOR, I2P presents an alternative with its own unique characteristics and potential advantages for specific anonymity requirements.

It is also important to note that threat actors can utilize both internal and external proxies 9. Compromised internal systems within a target network can be leveraged as **internal proxies**, allowing attacker traffic to blend in with legitimate network activity. Traffic originating from an internal IP address is often less likely to be flagged as malicious compared to connections from known attacker-controlled IPs. **External proxies**, often commercial services, provide readily available infrastructure for routing C2 traffic. These services offer ease of use and a broad range of IP addresses, although they might be subject to monitoring or takedown by law enforcement or security researchers.

|  |  |  |  |
| --- | --- | --- | --- |
| **Proxy Type** | **Purpose in C2 Infrastructure** | **Key Benefits for Attackers** | **Key Detection Challenges for Defenders** |
| Forward Proxy | Route traffic from attacker to target, mask origin IP | Anonymity of C2 server | Identifying the true source of traffic |
| Reverse Proxy | Hide C2 server location, load balancing, SSL termination | Enhanced resilience, obfuscation of server details | Discovering the actual backend server |
| Anonymizing Proxy | Obscure user/server identity by removing identifying info | Increased difficulty in tracing origin | Analyzing traffic for remaining identifiers |
| Rotating Proxy | Assign different IPs over time | Harder to track and block based on IP address | Detecting patterns across changing IPs |
| TOR Proxy | Route traffic through the TOR network for strong anonymity | High level of anonymity through multi-layered encryption and routing | Slow speed, potential for exit node monitoring |
| I2P Proxy | Route traffic through the I2P network for anonymity | Alternative anonymous network | Less mainstream, requires understanding of the I2P network |
| Internal Proxy | Route traffic through compromised internal systems | Blends in with legitimate network traffic | Distinguishing malicious activity from normal internal communications |
| External Proxy | Route traffic through commercial proxy services | Easy to use, wide range of IPs available | Service IPs may be known or monitored |

## Benefits of Proxy Servers for C2 Operations

The integration of proxy servers offers several key benefits for C2 operations. **Anonymity** is a primary advantage, as proxies can hide the true IP address of both the C2 server and the attacker 1. This makes attribution significantly more challenging and complicates efforts by defenders to directly target the attacker's infrastructure. If the attacker's IP address remains unknown, it becomes considerably harder to implement effective blocking measures at the network perimeter or pursue legal action.

Proxy servers also aid in **evading detection**. By routing C2 traffic through proxies, attackers can mask their malicious communications to appear as legitimate traffic 2. This can be achieved by utilizing common ports and protocols or by blending in with the network activity originating from compromised internal proxies. This technique can help bypass simple signature-based detection mechanisms and IP blacklisting strategies.

**Facilitating redirection** is another crucial benefit. Redirector servers, which are often proxy servers, are employed to hide the true location of the C2 server 9. These redirectors act as intermediary points, adding layers of indirection that make it more difficult for defenders to map the entire C2 infrastructure. If one redirector is identified and blocked, the attacker can quickly pivot to another without exposing the core C2 server.

In some instances, proxies can be leveraged to **bypass network controls**. Attackers might exploit misconfigurations or vulnerabilities in network firewalls or access control lists through proxy servers to reach resources that would otherwise be inaccessible 1. Tools like ProxyChains can be used to force any TCP connection made by an application to go through a specified proxy, potentially bypassing local authorized proxies and other network defenses 12.

Finally, **domain fronting** is a sophisticated evasion technique that relies on Content Delivery Networks (CDNs) to mask the actual destination of C2 traffic 9. This technique exploits the fact that the initial TLS handshake reveals a seemingly legitimate domain hosted on the CDN, while the traffic is actually directed to a different backend server controlled by the attacker. Domain fronting leverages the way CDNs route traffic based on the Host header in the HTTP request, allowing attackers to utilize high-reputation domains to conceal their malicious activity.

## Security Implications of Proxy Usage (Offensive and Defensive)

From an **offensive perspective**, the use of proxy servers in C2 infrastructure significantly increases stealth and reduces the risk of detection and attribution 9. Attackers can conduct operations from sources that appear benign or are difficult to trace back to them. The ability to utilize multiple proxies and redirectors enhances the resilience of the C2 infrastructure, making it more challenging for defenders to disrupt the communication channels.

Conversely, from a **defensive perspective**, proxy usage presents substantial challenges. Identifying the true source of malicious traffic becomes considerably more difficult 9. Blocking C2 communications without inadvertently disrupting legitimate traffic also poses a significant hurdle. Defenders need to employ advanced detection techniques to counter the obfuscation provided by proxies. This includes analyzing network traffic patterns for anomalies, monitoring for connections to known malicious proxy IP addresses, and potentially implementing SSL/TLS inspection to gain visibility into encrypted traffic 14.

Monitoring outbound connections for unusual destinations and patterns that deviate from normal organizational traffic can indicate potential C2 activity through proxies 7. Threat intelligence feeds that contain lists of known malicious proxy IP addresses and domains are also crucial for identifying and blocking such infrastructure 19. Tools like Zeek can analyze metadata from encrypted traffic, such as JA3 and JA3S fingerprints, without requiring decryption, to identify connections potentially associated with malicious actors or tools, including those utilizing proxies 20. If feasible and compliant with privacy regulations, deep packet inspection can reveal patterns and content indicative of C2 traffic, even when proxies are in use 7. Furthermore, monitoring endpoints for modifications of proxy configuration files, such as proxychains.conf, can alert defenders to unauthorized proxy usage by potential attackers 12.

## Chosen Proxy Solution and Justification

For the purpose of extending the C2 lab environment, a combination of open-source proxy solutions will be implemented to provide a comprehensive testing ground for various C2 scenarios. **Squid** will be utilized as a versatile forward and reverse proxy server 18. Its flexibility allows for simulating different proxy configurations, including basic proxying for traffic redirection and content filtering for mimicking certain defensive setups 18. Squid's ability to handle both HTTP and HTTPS traffic makes it suitable for a wide range of C2 communication protocols.

To simulate strong anonymity, **TOR** will be integrated into the lab 4. By using TOR and tools like ProxyChains, all network traffic from designated attacker machines can be routed through the TOR network, allowing for the study of highly anonymized C2 communications 4. This will enable experimentation with detection techniques specifically designed to identify TOR traffic.

Finally, **HAProxy** will be employed as a reverse proxy to enhance the resilience and stealth of the simulated C2 server infrastructure 31. HAProxy's load balancing capabilities will allow for simulating scenarios with multiple backend C2 servers, while its SSL termination features can be used to manage encrypted traffic in a controlled manner. This combination of Squid, TOR, and HAProxy offers a robust and flexible proxy infrastructure for the C2 lab, enabling the simulation of various attacker techniques and the evaluation of corresponding defensive strategies.

## Detailed Setup and Configuration of the Proxy Server(s)

### Squid Installation and Configuration:

1. Install Squid on a designated Linux VM using the distribution's package manager (e.g., sudo apt update && sudo apt install squid on Ubuntu) 21.
2. Open the main Squid configuration file, typically located at /etc/squid/squid.conf, using a text editor 18.
3. To configure Squid as a forward proxy, locate the http\_port directive and ensure it is set to a desired port (default is 3128). To allow connections from other machines in the lab, modify the http\_access deny all line to http\_access allow all for initial testing. For more restrictive access control, Access Control Lists (ACLs) can be defined based on source IP addresses 21.
4. To configure Squid as a reverse proxy, use directives like http\_port to specify the listening port and cache\_peer to define the backend C2 server's IP address and port 18.
5. Restart the Squid service after making any configuration changes (e.g., sudo systemctl restart squid on Ubuntu) 21.
6. Configure the attacker machine or the C2 platform to use the Squid proxy by setting the appropriate proxy address and port in the system's network settings or the C2 framework's configuration.

### TOR Installation and Configuration:

1. Install the TOR package on a designated Linux VM (e.g., sudo apt update && sudo apt install tor on Ubuntu) 4.
2. Start the TOR service (e.g., sudo systemctl start tor) 6.
3. To route traffic through TOR, install ProxyChains (e.g., sudo apt install proxychains) 4.
4. Edit the ProxyChains configuration file, typically located at /etc/proxychains.conf, and uncomment or add the socks4 127.0.0.1 9050 line at the end of the [ProxyList] section. This configures ProxyChains to use the local TOR SOCKS proxy running on port 9050 4.
5. To anonymize traffic from a specific application, prepend the proxychains command to the application's execution command (e.g., proxychains msfconsole to run Metasploit through TOR).

### HAProxy Installation and Configuration:

1. Install HAProxy on a designated Linux VM (e.g., sudo apt update && sudo apt install haproxy on Ubuntu) 31.
2. Open the HAProxy configuration file, typically located at /etc/haproxy/haproxy.cfg, using a text editor 32.
3. Configure a frontend section to define how HAProxy listens for connections. For example, to listen on all interfaces on port 443 for HTTPS traffic:  
   frontend c2\_frontend  
       bind \*:443  
       mode tcp  
       default\_backend c2\_backend
4. Configure a backend section to define the C2 server(s) to which HAProxy forwards traffic. For example, to forward traffic to a C2 server at IP address 192.168.1.100 on port 443:  
   backend c2\_backend  
       mode tcp  
       server c2\_server 192.168.1.100:443 check
5. Restart the HAProxy service after making configuration changes (e.g., sudo systemctl restart haproxy) 32.
6. Configure the registered domain's DNS records to point to the IP address of the HAProxy server.

# 3. DNS Infrastructure Implementation

## Purpose and Necessary Components of Setting Up a DNS Infrastructure for a C2 Lab Environment

The Domain Name System (DNS) is a fundamental component of internet communication, responsible for translating human-readable domain names into the IP addresses that computers use to communicate 35. In a C2 lab environment, establishing a custom DNS infrastructure can significantly enhance the realism and capability of simulated operations.

The necessary components for setting up a DNS infrastructure include **DNS server software**, such as BIND9 (Berkeley Internet Name Domain) 37. BIND9 is a widely used, open-source DNS server that offers extensive configuration options suitable for a penetration testing lab. Another crucial component is the **authoritative zone configuration**. This involves setting up DNS zones for the domains that will be used within the lab environment, including the registered domain for the C2 server and potentially internal domains for simulated victim machines 37. Finally, **resolver configuration** is required, which involves configuring the lab machines (both attacker and victim VMs) to use the custom DNS server(s) for name resolution 37.

When considering options, a **local DNS server** offers significant advantages for a C2 lab 37. It provides complete control over DNS resolution within the lab, allowing for the simulation of internal domains and specific DNS behaviors that are critical for testing various C2 techniques. Relying solely on existing external DNS services would lack the necessary flexibility to simulate certain C2-related DNS activities, such as DNS tunneling over custom domains, as external servers would not have records for internal lab domains.

## Explain How Having a Custom DNS Setup Can Enhance the Realism and Capability of C2 Operations

A custom DNS setup significantly enhances the realism and capability of C2 operations within a lab environment. It allows for **simulating real-world domain resolution**, accurately mimicking how domain names are translated into IP addresses in a live network. This is crucial for testing the end-to-end communication flow of a C2 channel.

Furthermore, a custom DNS infrastructure facilitates more **covert communication** techniques. **DNS tunneling**, a method of encoding commands and data within DNS queries and responses, can be effectively simulated 9. DNS traffic is often permitted through firewalls, making it a potentially covert channel for C2 communication. By controlling the DNS server, the lab environment can be configured to properly handle and analyze this type of traffic. Similarly, **DNS beaconing**, where infected hosts periodically send DNS queries to a C2 server to check for commands, can be realistically simulated 44.

Finally, a custom DNS setup allows for **testing DNS-based evasion techniques**. Scenarios where attackers might manipulate DNS records or use DNS to bypass security controls can be created and analyzed within the controlled lab environment 46.

## Discuss the Offensive and Defensive Security Implications of Managing DNS Within a C2 Infrastructure

From an **offensive perspective**, managing DNS within a C2 infrastructure enables the establishment of covert communication channels that are often more difficult to detect than traditional network protocols. The potential for **data exfiltration over DNS** also exists 9. Additionally, attackers can use their own DNS server to resolve attacker-controlled domains that might be blocked by standard DNS filtering mechanisms, providing a more resilient communication pathway.

From a **defensive perspective**, managing DNS by an attacker presents several challenges. Distinguishing malicious DNS traffic from legitimate DNS queries can be difficult. Defenders require specialized DNS monitoring and analysis tools to detect anomalies and potential DNS tunneling or beaconing 35. Monitoring DNS query patterns for **high entropy domains** can be an indicator of potential DNS tunneling, as encoded data often results in long, seemingly random domain names 38. Analyzing DNS traffic for unusual frequencies or connections to specific destinations can also reveal C2 activity 35. Implementing **DNS filtering** and blocking known malicious DNS servers and domains is a crucial defensive measure 7. Furthermore, **protective DNS solutions** that analyze DNS traffic in real-time and block malicious domains based on threat intelligence and behavioral analysis can provide an additional layer of security at the DNS level 35.

## Chosen DNS Solution and Justification

For the C2 lab environment, **BIND9** has been chosen as the DNS server software 37. Its robust feature set, extensive configurability, and wide adoption make it an ideal choice for simulating various DNS-related attacks and defenses. BIND9 allows for the creation of authoritative zones, the configuration of various DNS record types, and the implementation of different DNS resolution behaviors, providing the necessary flexibility for a realistic C2 lab. A **local, authoritative DNS server** is preferred for this lab environment as it grants complete control over the DNS namespace used for testing, enabling the simulation of both internal and external domain resolution relevant to C2 operations.

## Detailed Setup and Configuration of the DNS Infrastructure

1. Install BIND9 on a designated Linux VM using the package manager (e.g., sudo apt update && sudo apt install bind9 on Ubuntu) 37.
2. Configure BIND9 to listen on the appropriate network interface for the lab environment by editing the /etc/bind/named.conf.options file. Ensure that the listen-on directive specifies the correct IP address(es) of the DNS server. For a basic lab setup, you might allow queries from any source using allow-query { any; };, but for more controlled scenarios, restrict this to the lab's network range 37.
3. Create an authoritative DNS zone for the registered domain and any other lab domains by editing the /etc/bind/named.conf.local file. Add a zone entry for each domain, specifying the zone name, type (master), and the file where the zone's records will be stored (e.g., /etc/bind/db.labdomain.com) 37.
4. Create the zone files (e.g., /etc/bind/db.labdomain.com) based on the /etc/bind/db.local template. These files will contain the DNS records for the domain, including the Start of Authority (SOA) record, Name Server (NS) record pointing to the DNS server itself, and A records mapping hostnames to IP addresses. For the registered C2 domain, create an A record that points the domain to the IP address of the C2 server (or the HAProxy server if used as a reverse proxy) 37. Also, create any necessary A records for simulated victim machines within the lab's network.
5. Configure the lab machines (attacker and victim VMs) to use the newly set up DNS server by modifying their network settings to point to the DNS server's IP address 37.
6. Reload the BIND service after making any configuration changes (e.g., sudo systemctl reload bind9) 37.
7. Verify the DNS setup by using tools like nslookup on the lab machines to ensure that the registered domain and other lab hostnames resolve to the correct IP addresses 37.

# 4. Registered Domain Integration

## Research the Process of Registering a Domain Name for Use in a C2 Lab Environment

The process of registering a domain name involves interacting with domain name registrars, which are organizations accredited by the Internet Corporation for Assigned Names and Numbers (ICANN) to sell and manage domain names 49. When choosing a registrar for a C2 lab environment, several considerations are important. **Cost** is a factor, as prices can vary between registrars. **Privacy policies** should be reviewed, particularly regarding the handling of personal information provided during registration. **Security features** offered by the registrar, such as domain transfer locks and two-factor authentication, are also crucial 49. Finally, the **ease of use** of the registrar's website and domain management interface should be considered. Opting for a registrar accredited by ICANN is generally recommended for enhanced security and adherence to established standards 49. It is also advisable to consider registrars that offer WHOIS privacy protection services, which can obscure the personal contact information associated with the domain registration in the public WHOIS database, helping to protect the operator's identity 49.

The information typically required for domain registration includes contact details for the registrant (name, address, phone number, email address), as well as payment information 51. Even for lab purposes, it is prudent to use non-identifying information where possible to maintain a degree of operational security 49. This might involve using a generic organization name and a dedicated email address not directly linked to personal identities.

## Explain How Using a Registered Domain (Even if for Lab Purposes) Enhances the Realism and Capability of C2 Operations

Utilizing a registered domain, even within the confines of a lab environment, significantly enhances the realism and capability of simulated C2 operations. C2 traffic that uses a registered domain appears more legitimate to network monitoring tools and security analysts compared to traffic relying solely on IP addresses or non-existent domains. This increased realism allows for a more accurate assessment of detection capabilities.

Furthermore, newly registered domains might not yet be included in basic domain blacklists used by some security services 52. Threat actors often register domains shortly before launching attacks to evade initial detection based on domain reputation 52. By using a registered domain in the lab, this aspect of threat actor behavior can be simulated and studied.

Finally, having a registered domain enables the testing of domain-based C2 techniques, such as domain fronting, which inherently rely on the existence of a valid, registered domain name 10.

## Discuss the Associated Security Implications, Such as the Potential for Detection Based on Domain Reputation

Despite being used for lab purposes, a registered domain carries associated security implications. **Domain reputation** is a key factor. If the domain is consistently used in simulated suspicious or malicious activities within the lab, it could eventually be flagged by security services and threat intelligence platforms based on its history 52. Security solutions often track the reputation of domains, and repeated association with C2 activity is likely to lead to a negative reputation over time.

The **WHOIS information**, even if protected by a privacy service, could potentially become a point of investigation if the domain's activity is deemed sufficiently malicious or suspicious. While privacy services obscure the details, the fact that such a service is used might itself be noted.

Moreover, the details of the domain registration, such as the registrar used and the registration date, could potentially be correlated with other indicators of compromise to identify broader threat actor infrastructure 53. For example, a sudden registration of a domain with specific characteristics might be linked to known attack campaigns.

# 5. Component Interaction with Existing C2 Platforms: Integrating Proxies, DNS, and the Registered Domain with Metasploit, Cobalt Strike, and Empire

To effectively utilize the integrated proxy servers, custom DNS infrastructure, and registered domain within the C2 lab, each existing C2 platform needs to be configured accordingly.

**Metasploit:** To configure Metasploit to use the proxy servers, global or local proxy options can be set within the msfconsole 28. The setg command can be used to set a global proxy, while the set command can configure a proxy for a specific module. The http\_proxy and https\_proxy options can be set to the address and port of the chosen Squid or TOR proxy server. For listeners, the LHOST option should be set to the IP address of the machine running Metasploit, and the registered domain can be used in conjunction with the custom DNS server to simulate real-world scenarios. Payloads can also be configured to connect back to the registered domain.

**Cobalt Strike:** Cobalt Strike offers several ways to integrate proxies 29. Listeners can be configured to use specific protocols (HTTP, HTTPS, DNS) and can be associated with the registered domain. For using TOR, Cobalt Strike can be integrated with ProxyChains by configuring ProxyChains to route traffic through the TOR SOCKS proxy. Cobalt Strike also has a built-in SOCKS proxy server that can be enabled, allowing other tools to pivot through a compromised host. The connect command can be used to specify a direct connection through a proxy. The registered domain should be specified in the listener's Host field. The custom DNS server will ensure that this domain resolves to the Cobalt Strike team server's IP address.

**Empire:** Empire is designed to be a proxy-aware C2 framework 56. Agents can be configured to communicate through proxy servers. The configuration typically involves setting the Proxy option for a listener to the address and port of the proxy server. The registered domain should be used when setting up listeners, particularly HTTP/HTTPS listeners, by specifying it in the Host option. The custom DNS server will then resolve this domain to the Empire C2 server's IP address.

In all cases, the custom DNS infrastructure will play a crucial role in resolving the registered domain to the appropriate IP address of the C2 server (or the HAProxy server acting as a reverse proxy). This ensures that when the C2 platforms attempt to establish listeners or when payloads attempt to connect back, the registered domain is correctly translated to the underlying infrastructure. For C2 servers hosted outside the lab network, the local DNS server can be configured to forward requests for the registered domain to external resolvers after resolving any internal lab hostnames.

# 6. Network Diagram of the Extended C2 Lab Environment

A computer diagram of a computer

AI-generated content may be incorrect.

* **Attacker Machine(s):** Running Metasploit, Cobalt Strike, and Empire, configured to use proxy servers.
* **Proxy Server(s):** Squid (forward/reverse), TOR, HAProxy, showing traffic flow.
* **C2 Server(s):** Hosting the C2 platforms, associated with the registered domain.
* **Custom DNS Server:** Resolving the registered domain and lab-internal hostnames.
* **Registered Domain:** Pointing to the C2 server's public IP (or HAProxy).
* **Victim Machine(s):** Simulating compromised hosts within the lab network, configured to use the custom DNS server.

# 7. Comprehensive Security Implications and Mitigations

## Offensive Security Perspective

The extended C2 lab environment significantly enhances the attacker's capabilities. The integration of proxy servers provides a realistic way to simulate **anonymity**, making it harder to trace the origin of simulated attacks. The use of redirectors further adds to the **evasion** capabilities by obscuring the true location of the C2 infrastructure. The custom DNS infrastructure allows for the simulation of **covert communication channels** through techniques like DNS tunneling and beaconing, which can bypass traditional network monitoring. Finally, the use of a registered domain adds a layer of **realism** to the C2 operations, allowing for the testing of techniques like domain fronting and the study of how domain reputation might affect detection.

## Defensive Security Perspective and Detection Strategies

Defenders face a more complex challenge in detecting C2 activity within an environment utilizing proxies, custom DNS, and registered domains 3.

**Proxy Servers:** Detection strategies include **monitoring outbound connections** for unusual patterns or connections to known proxy services, especially on non-standard ports 12. **Analyzing network traffic** for characteristics of proxy protocols (e.g., CONNECT requests) can also be effective. Utilizing **threat intelligence feeds** that contain lists of known malicious proxy IPs is crucial for identifying and blocking suspicious connections.

**Custom DNS:** Detecting C2 activity through custom DNS involves **analyzing DNS query patterns** for anomalies, such as a high volume of requests to a single domain or requests with unusually long or seemingly random subdomains, which could indicate DNS tunneling 35. Monitoring for **resolutions of newly registered or low-reputation domains** can also be an indicator of potential C2 infrastructure setup. Specialized tools and techniques for detecting **DNS tunneling and beaconing** should be employed.

**Registered Domain:** Detection related to the registered domain includes **monitoring the domain's reputation** through threat intelligence services. Analyzing network traffic associated with the domain for **anomalous patterns** or communication with unusual IP addresses can also be indicative of malicious activity.

**Mitigation Strategies for Each Component**

**Proxy Servers:** Mitigation strategies include **implementing strict egress filtering** on firewalls to limit outbound connections to only necessary services and known good destinations. **Blocking connections to known malicious proxy IPs** identified through threat intelligence is essential. Organizations might also consider **SSL/TLS inspection** to gain visibility into encrypted traffic, although this must be implemented with careful consideration of privacy implications.

**Custom DNS:** Mitigation involves **implementing DNS filtering** to block resolutions of known malicious domains and **sinkholing** attempts to reach C2 infrastructure 7. **Monitoring DNS traffic** for anomalies and potentially using **protective DNS services** that provide real-time threat analysis and blocking can also be effective.

**Registered Domain:** Mitigation includes **monitoring the reputation of domains** being accessed by internal hosts and **blocking access to newly registered or domains with suspicious reputations** 52. Analyzing network traffic associated with the domain for unusual patterns can also aid in detection and mitigation.

Implementing **SSL/TLS inspection** within the lab environment can provide a significant advantage in analyzing C2 traffic, even when encryption is used 16. This allows for the inspection of the content of HTTPS traffic, which is commonly used for C2. However, this requires careful setup, certificate management, and consideration of potential performance impacts.

|  |  |  |  |
| --- | --- | --- | --- |
| **Component** | **Key Offensive Security Implications** | **Key Defensive Security Implications** | **Mitigation Strategies** |
| Proxy Server | Increased anonymity, evasion of IP-based blocking, traffic redirection | Difficulty in identifying true source, blending with legitimate traffic | Egress filtering, blocking known malicious proxies, SSL/TLS inspection |
| Custom DNS | Covert communication (tunneling, beaconing), bypassing DNS filtering | Distinguishing malicious DNS traffic, detecting covert channels | DNS filtering, sinkholing, monitoring for anomalous patterns, protective DNS |
| Registered Domain | Realistic C2 appearance, potential for domain fronting | Detection based on domain reputation, correlation with other indicators | Monitoring domain reputation, blocking suspicious domains, analyzing associated network traffic for anomalies |

# 8. Potential Challenges and Risks in Implementation

Implementing these extensions to the C2 lab environment may present several challenges. The **complexity of setting up and configuring multiple network components** (proxy servers, DNS server, C2 platforms) can be significant and may require a solid understanding of networking principles. There is a **potential for configuration errors** that could impact the functionality or the realism of the lab, requiring careful troubleshooting. Running multiple virtual machines and services can also lead to **resource constraints** on the underlying hardware. When using a registered domain, even for lab purposes, it is important to be mindful of **ethical considerations** and ensure that the domain's usage is clearly for research and testing to avoid any unintended negative consequences. Finally, the cybersecurity landscape is constantly evolving, so **keeping the lab environment up-to-date** with the latest C2 techniques and defensive measures will require ongoing effort.

# 9. Conclusion

Extending a C2 lab environment with proxy servers, a custom DNS infrastructure, and a registered domain offers significant value for enhancing cybersecurity skills and gaining a deeper understanding of real-world threat actor tactics. The integration of these components allows for the simulation of more sophisticated attack scenarios, including those involving anonymity, evasion, and covert communication. While the implementation may present certain challenges, the resulting environment provides a more realistic and capable platform for training, research, and the development of advanced security tools and detection methodologies. By carefully considering both the offensive and defensive security implications of these extensions, cybersecurity professionals can leverage this enhanced lab environment to better prepare for and respond to the evolving threat landscape.

# References

1. Uses of Proxy servers and different types of proxy servers, accessed march 21, 2025, <https://www.securiwiser.com/blog/different-types-of-proxy-servers-and-their-uses/>
2. What Is a Proxy Server Used For? (And How Does It Work?) - YouTube, accessed march 21, 2025, <https://www.youtube.com/watch?v=017eLXeVuj0>
3. MITRE ATT&CK Matrix - C2 Connection Proxy - Active Countermeasures, accessed march 21, 2025, <https://www.activecountermeasures.com/mitre-attck-matrix-c2-connection-proxy/>
4. Anonymize Your Traffic With Proxychains & Tor: A Comprehensive Guide - Spyboy blog, accessed march 21, 2025, <https://spyboy.blog/2024/01/02/anonymize-your-traffic-with-proxychains-tor-a-comprehensive-guide/>
5. How to set up a proxy for Tor - IPRoyal.com, accessed march 21, 2025, <https://iproyal.com/blog/tor-proxy/>
6. How to stay anonymous during a penetration test (with Tor and ProxyChains) - YouTube, accessed march 21, 2025, <https://www.youtube.com/watch?v=LXmF8hJlbVE>
7. Using the Tor api to make an anonymous proxy server - Stack Overflow, accessed march 21, 2025, <https://stackoverflow.com/questions/17976281/using-the-tor-api-to-make-an-anonymous-proxy-server>
8. How to be Anonymous?. Installing & Configuring TOR with… | by Nidhi Rohilla - Medium, accessed march 21, 2025, <https://medium.com/@nidhirohilla.nr/how-to-be-anonymous-7f8651c5dbe6>
9. What is Command and Control (C&C or C2) in Cybersecurity? - zenarmor.com, accessed march 21, 2025, <https://www.zenarmor.com/docs/network-security-tutorials/what-is-command-and-control-c2>
10. Proxy, Technique T1090 - Enterprise | MITRE ATT&CK®, accessed march 21, 2025, <https://attack.mitre.org/techniques/T1090/>
11. Havoc C2 with AV/EDR Bypass Methods in 2024 (Part 1) | by Sam Rothlisberger - Medium, accessed march 21, 2025, <https://medium.com/@sam.rothlisberger/havoc-c2-with-av-edr-bypass-methods-in-2024-part-1-733d423fc67b>
12. Walking Through Walls: Four Common Endpoint Tools Used to ..., accessed march 21, 2025, <https://www.crowdstrike.com/en-us/blog/4-common-endpoint-tools-used-to-facilitate-covert-c2/>
13. Cobalt Strike, a Defender's Guide – Part 2 - The DFIR Report, accessed march 21, 2025, <https://thedfirreport.com/2022/01/24/cobalt-strike-a-defenders-guide-part-2/>
14. What is C2? Command and Control Infrastructure Explained - Varonis, accessed march 21, 2025, <https://www.varonis.com/blog/what-is-c2>
15. (QR) Coding My Way Out of Here: C2 in Browser Isolation Environments - Google Cloud, accessed march 21, 2025, <https://cloud.google.com/blog/topics/threat-intelligence/c2-browser-isolation-environments>
16. What is TLS Inspection - Check Point Software, accessed march 21, 2025, <https://www.checkpoint.com/cyber-hub/network-security/what-is-network-security/what-is-tls-inspection/>
17. Encrypted Channel, Technique T1573 - Enterprise - MITRE ATT&CK®, accessed march 21, 2025, <https://attack.mitre.org/techniques/T1573/>
18. Squid Proxy Server 3.1: Beginner's Guide | Cloud & Networking | Paperback - Packt, accessed march 21, 2025, <https://www.packtpub.com/en-ee/product/squid-proxy-server-31-beginners-guide-9781849513906>
19. Active C2 Servers - VulnCheck, accessed march 21, 2025, <https://vulncheck.com/blog/active-c2-servers>
20. How to Spot C2 Traffic on Your Network - YouTube, accessed march 21, 2025, <https://www.youtube.com/watch?v=N43sfHATDIk>
21. How to Install and Configure Squid Proxy on Ubuntu - phoenixNAP, accessed march 21, 2025, <https://phoenixnap.com/kb/setup-install-squid-proxy-server-ubuntu>
22. Squid Proxy Server 3.1: Beginner's Guide | Cloud & Networking | Subscription - Packt, accessed march 21, 2025, <https://www.packtpub.com/en-gb/product/squid-proxy-server-31-beginners-guide-9781849513906?type=subscription>
23. How to Set Up Squid Proxy for Private Connections on Ubuntu 20.04 - DigitalOcean, accessed march 21, 2025, <https://www.digitalocean.com/community/tutorials/how-to-set-up-squid-proxy-on-ubuntu-20-04>
24. What is Squid Proxy Server? | How to Configure & Setup? - ManageEngine Firewall Analyzer, accessed march 21, 2025, <https://www.manageengine.com/products/firewall/tech-topics/what-is-squid-proxy.html>
25. Squid 2.5.x/3.x - NTLM Buffer Overflow (Metasploit) - Multiple remote Exploit, accessed march 21, 2025, <https://www.exploit-db.com/exploits/9951>
26. Squid NTLM Authenticate Overflow - Rapid7, accessed march 21, 2025, <https://www.rapid7.com/db/modules/exploit/linux/proxy/squid_ntlm_authenticate/>
27. Squid Proxy Port Scanner - Rapid7, accessed march 21, 2025, <https://www.rapid7.com/db/modules/auxiliary/scanner/http/squid_pivot_scanning/>
28. The ins and outs of HTTP and HTTPS communications in Meterpreter and Metasploit Stagers, accessed march 21, 2025, <https://docs.metasploit.com/docs/using-metasploit/advanced/meterpreter/the-ins-and-outs-of-http-and-https-communications-in-meterpreter-and-metasploit-stagers.html>
29. Cobalt Strike | Hacknetics - GitBook, accessed march 21, 2025, <https://ice-wzl.gitbook.io/hacknetics/c2-frameworks/cobalt-strike>
30. RedTeaming\_CheatSheet/cobalt-strike.md at main - GitHub, accessed march 21, 2025, <https://github.com/0xJs/RedTeaming_CheatSheet/blob/main/cobalt-strike.md>
31. Installing and configuring HAProxy - Kaspersky, accessed march 21, 2025, <https://support.kaspersky.com/kwts/6.0/en-US/167740.htm>
32. Advanced HAProxy Configurations for High Availability and Performance Optimization - LoadForge Guides, accessed march 21, 2025, <https://loadforge.com/guides/advanced-haproxy-configurations-for-high-availability>
33. How to Use HAProxy for Load Balancing | Linode Docs, accessed march 21, 2025, <https://www.linode.com/docs/guides/how-to-use-haproxy-for-load-balancing/>
34. Load balancing with HAProxy - Percona XtraDB Cluster, accessed march 21, 2025, <https://docs.percona.com/percona-xtradb-cluster/5.7/howtos/haproxy.html>
35. UltraDDR is Next-Level Protective DNS. Here's Why. - Vercara, accessed march 21, 2025, <https://vercara.com/resources/ultraddr-is-next-level-protective-dns-heres-why>
36. What is DNS? - A DNS Security Primer - Center for Cybersecurity Policy and Law, accessed march 21, 2025, <https://www.centerforcybersecuritypolicy.org/insights-and-research/what-is-dns---a-dns-security-primer>
37. Set up a Simple DNS Server in the Lab — StarlingX documentation, accessed march 21, 2025, <https://docs.starlingx.io/deploy_install_guides/release/setup-simple-dns-server-in-lab.html>
38. Threat Hunting: DNS C2. DNS is the protocol used in internal… | by ..., accessed march 21, 2025, <https://medium.com/@lsepaolo/threat-hunting-dns-c2-c461abf4794d>
39. C2 Communication Using DNS. TL;DR — C2 communication over ..., accessed march 21, 2025, <https://medium.com/@0x0vid/creating-a-dns-c2-server-and-agent-26fb8d556dc0>
40. Lab Building Guide: Virtual Active Directory | by Vartai Security - Medium, accessed march 21, 2025, <https://medium.com/@vartaisecurity/lab-building-guide-virtual-active-directory-5f0d0c8eb907>
41. Building A Test Lab For Pentesting - Cyber Advisors Blog, accessed march 21, 2025, <https://blog.cyberadvisors.com/technical-blog/blog/building-a-test-lab-for-pentesting>
42. Setting Up a VPN-Based Penetration Testing Lab | Infosec, accessed march 21, 2025, <https://www.infosecinstitute.com/resources/penetration-testing/setting-up-a-vpn-based-penetration-testing-lab/>
43. Entry Level Pentesting Lab Walkthrough - YouTube, accessed march 21, 2025, <https://www.youtube.com/watch?v=C3Eg8QPPWcY>
44. DNS-Layer Security: The Ultimate Guide to What It Is and Why You Need It - Cisco Umbrella, accessed march 21, 2025, <https://umbrella.cisco.com/blog/what-is-dns-layer-security>
45. Stop Attackers from Using DNS Against You - Palo Alto Networks, accessed march 21, 2025, <https://www.paloaltonetworks.com/resources/whitepapers/stop-attackers-from-using-dns-against-you.viewer.html>
46. The Most Common DNS Security Risks in 2025 (And How to Mitigate Them), accessed march 21, 2025, <https://heimdalsecurity.com/blog/dns-security-risks/>
47. The Secret World of DNS Tunneling: Understanding the Risks - Enterprise Defence, accessed march 21, 2025, <https://enterprisedefence.com/blog/dns-tunneling-understanding-the-risks/>
48. The DNS Line of Defense When Others Fail - HYAS, accessed march 21, 2025, <https://www.hyas.com/blog/the-dns-line-of-defense-when-others-fail>
49. Cybersecurity Due Diligence: Secure Domain Name Registrars, accessed march 21, 2025, <https://www.encirca.com/blog/cybersecurity-due-diligence-secure-domain-name-registrars/>
50. Automated Penetration Testing: A Guide to Domain Collection - Resource Detail - TECNO Mobile, accessed march 21, 2025, <https://security.tecno.com/SRC/blogdetail/377?lang=en_US>
51. Domain registration details - Web Penetration Testing with Kali Linux - Third Edition [Book], accessed march 21, 2025, <https://www.oreilly.com/library/view/web-penetration-testing/9781788623377/900ee04d-2254-42b1-b21a-61a52f57d3f7.xhtml>
52. Why Are Newly Registered Domains Important for Cybersecurity? - WhoAPI Inc., accessed march 21, 2025, <https://whoapi.com/blog/why-are-newly-registered-domains-important-for-cybersecurity/>
53. APT Groups Registering C2 Domains Way Before Attacks | Cyware Alerts - Hacker News, accessed march 21, 2025, <https://social.cyware.com/news/apt-groups-registering-c2-domains-way-before-attacks-e8eb5cb9>
54. CobaltStrike Guide. #5 Browser pivoting and SSH sessions - HackYourMom, accessed march 21, 2025, <https://hackyourmom.com/en/pryvatnist/posibnyk-z-cobaltstrike-%E2%84%965-browser-pivonting-ta-ssh-sesiyi/>
55. Reverse Port Forward through a SOCKS Proxy - Cobalt Strike, accessed march 21, 2025, <https://www.cobaltstrike.com/blog/reverse-port-forward-through-a-socks-proxy>
56. PoshC2: Proxy-Aware Cross-Platform C2 Framework - Hunt.io, accessed march 21, 2025, <https://hunt.io/malware-families/posh-c2>
57. C2 Hunting: How To Find C2 Servers With Shodan - Kraven Security, accessed march 21, 2025, <https://kravensecurity.com/c2-hunting-using-shodan/>
58. 3.6 Android Pen testing Basics - Intercepting mobile network traffic - YouTube, accessed march 21, 2025, <https://www.youtube.com/watch?v=r0NqCzAx-_o>
59. SSL/TLS Implementation Best Practices - zenarmor.com, accessed march 21, 2025, <https://www.zenarmor.com/docs/network-security-tutorials/best-practices-for-ssl-tls-implementation>
60. Using SSL/TLS certificates with TLS inspection configurations in AWS Network Firewall, accessed march 21, 2025, <https://docs.aws.amazon.com/network-firewall/latest/developerguide/tls-inspection-certificate-requirements.html>
61. TLS Inspection Guide: Understanding, Implementation, and Considerations - Zenarmor, accessed march 21, 2025, <https://www.zenarmor.com/docs/network-security-tutorials/what-is-tls-inspection>
62. SSL Inspection Deployment and Operations Guide - Zscaler Help Portal, accessed march 21, 2025, <https://help.zscaler.com/zscaler-deployments-operations/ssl-inspection-deployment-and-operations-guide>
63. The Ultimate Guide to SSL/TLS Decryption - Cyber Defense Magazine, accessed march 21, 2025, <https://www.cyberdefensemagazine.com/the-ultimate-guide/>
64. How to inspect TLS encrypted traffic - APNIC Blog, accessed march 21, 2025, <https://blog.apnic.net/2024/10/14/how-to-inspect-tls-encrypted-traffic/>
65. Extending C2 Traffic Detection Methodologies: From TLS 1.2 to TLS 1.3-enabled Malware | Request PDF - ResearchGate, accessed march 21, 2025, <https://www.researchgate.net/publication/384463247_Extending_C2_Traffic_Detection_Methodologies_From_TLS_12_to_TLS_13-enabled_Malware>